# Mathematical Model Reduction Principle Development

Yaroslav Matviychuk, Nataliia Shakhovska

*Lviv Polytechnic National University, Lviv, 79013, Ukraine.*

**Abstract**

The new principle of removing elements of mathematical model based on its parametric identification is proposed in the paper. It allows reducing computational and time complexity of the applications built on the model. The computation complexity reducing is important for online data processing. Example of such system is medical decision support system. Also, IoT-based solutions can used the proposed approach. In addition, reducing complexity can increase the accuracy of mathematical models. Therefore, reducing the number of parameters is an important step in data preprocessing, which is used in almost all modern systems. Known methods of reducing the dimension depend on the problem area, which makes it impossible to use them in ensemble models. The proposed method for the reduction of mathematical models is used for the ordinary differential equations and on the neural network model.

**Keywords 1**

mathematical model, reduction, identification procedure, incorrectness, neural network, ordinary differential equation (ODE).

## 1. Introduction

An identification of a mathematical model is reversed problem, and therefore incorrect. Identification errors can occur from redundant model elements. This is used to detect unnecessary elements. Especially this is important for big data processing in real time (medicine, smart city etc.) [1].

The dynamic object can be represented in the differential equations form. That is why the complexity decreasing is given the model simplifying. In most of these methods, the problem of taking into account the total error in assessing the quality indicators of the systems is not posed.

Other examples of complex system are neural networks, particularly deep neural networks, auto-encoders and predictive models. For this purpose, the dimensionality reduction is widely used.

The novel reduction principle of the mathematical model is presented in the paper. This principle is used for a long time in a simplified form to reduce mathematical models in the form of ordinary differential equations.

The novelty of the proposed principle is the following:

- simpler identifying of redundant parameters based on relative deviations;
- removing of the elements with biggest relative deviations within the model.

This allows increasing the accuracy of identification of mathematical model. The algorithm is processed for models in the forms of neural network and differential equations.

The paper consists of following sections. Section 2 Related works presents methods of model complexity reduction. In section 3 the reduction method is given. Section 4 presents the reduction method implementation for different domains. The last section concludes this paper containing the probable decision of appraisal technique.

IDDM'2020: 3rd International Conference on Informatics & Data-Driven Medicine, November 19–21, 2020, Växjö, Sweden
EMAIL: matv@ua.fm (A.1), natalya233@gmail.com (A.2);
ORCID: 0000-0002-5570-182X (A.1), 0000-0002-6875-8534 (A.2)

## 2. Related works

Analysis of methods for simplifying mathematical models of dynamic systems shows the use of two main approaches: construction of a simplified model based on the criterion of proximity of quality indicators of the original and simplified models in the image space and in the state space [2].

The paper [3] presents uniform manifold approximation and projection (UMAP) reduction based on Riemannian geometry and algebraic topology. The main disadvantage of this method is time complexity.

In [4] there is proposed the locality preserving projections algorithm. However, this algorithm can be used only for linear dimensionality reduction.

The idea of reducing redundant elements in machine learning algorithms is not new at all. For, example, PCA (Principal Component Analysis) and k-means algorithms are used for dimensionality reduction in [5]. This classical approach allows also finding outliers. The disadvantage of these algorithms is ability to process only discrete data.

Zuo Z. [6] proposed multi-agent approach and identified using Lyapunov–Krasovskii functional in the time domain. As it is shown in the experimental results, the quality of the method depends on the domain.

The model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders is used in [7]. The proposed method is simplified the model using the projection on nonlinear manifolds. The similar results are obtained in [8] too.

The reduction is widely used in neural networks too. The basic idea is to find a subset of useless weights on the network and set them to zero. Without exhaustive search, it is hard to tell which weights are involved in the prediction. That is why such type of reduction depends on the model's complexity. For reduction of unnecessary elements, Arnold – Kolmogorov – Hecht – Nielsen theorem is used [9, 13]. In this sense, the reduction of connections can be compared with the method of shutting down random neurons (dropout) during network training. In addition, if there are many zeros in the network, it takes up less space in the archive and can be read faster on some architectures.

Group LASSO regularization is most often used to keep useless weights in networks close to zero. Unlike regular regularization, we do not regularize layer weights or activations directly [10]. However, there are difficulties with the application of channel reduction to "branched" architectures and residual-networks (ResNet). After trimming the extra neurons during the merging of the branches, the dimensions may not match.

Using the variation optimization [11], we can approximate the discrete distribution of zeros and ones in the masking layer with a continuous one and optimize the parameters of the latter using the usual backpropagation algorithm. The main disadvantage is the dependence of empirical hyperparameters.

It is much more interesting to throw out not individual weights, but neurons from fully connected layers or channels from convolutions as a whole. In this case, the effect of compressing the network and speeding up predictions is much more pronounced. For this purpose, the other structure of neural network can be used.

The papers [12, 13] present the Neural-Like Structures based on Geometric Data Transformations. The main advantages of the proposed method are the following: not iterative training process, the high performance in training process, which creates conditions for solution of large-dimension tasks. This approach allows the time complexity reduction, but the number of model's parameters is the same.

The paper [14] proposed GMDH-neuro-fuzzy system with small number of hyperparameters but with huge time complexity.

The pruning technic can be used for the neural networks reducing too. The variational Bayesian scheme for pruning convolutional neural networks in channel level is proposed in [15]. The channel pruning without desire of re-training stage allows the computation complexity reducing.

Thus, the biggest part of the reduction methods depends on the domain. That is why the proposed method takes into account the error rate and can be used for different models.

The reduction principle can be used for software reliability modeling too. Accurate software valuation models can greatly help software project managers: project managers will be able to make

informed decisions on resource management, administration and planning project, and as a result will be able to complete the project on time and within the planned budget, which is a problem today.

The essence of the proposed method is to find a functional subset with less variability results and higher accuracy than for the initial functional set of the model [16]. In this case the functional set includes the parameters of the model that allow it to be calibrated.

# 3. The materials and methods

## 3.1. An Exploration of a Reduction Principle Implemented

For a simulated object there is an exact mathematical model with known parameters ($p_1$,…, $p_n$), which are the numerical values of the model elements. The model parameters are calculated using an identification procedure based on some data.

The mathematical model and the identification procedure must satisfy the following two conditions.

1. If the parameter is 0, it means the absence of a model element.
2. The model parameters continuously depend on the identification data within some environment of this data.

The parameter ($p_1$,…, $p_n$) is now will extended with additional parameters ($p_1$,…, $p_n$, $p_{n+1}$,…, $p_m$). So, the identification procedure for redundant parameters ($p_{n+1}$,…, $p_m$) will calculate the values close to zero (condition 1, within the accuracy of calculations):

$$p_i \approx 0; \;\; i = n+1,\ldots,m. \tag{1}$$

However, this property cannot detect unnecessary parameters, because some of necessary parameters may be close to zero.

Then we introduce perturbations, named *disturb*, to the data of identification, but within the environment of continuously (given as condition 2). The identification procedure will calculate the parameters ($p_1$',…, $p_n$', $p_{n+1}$',…, $p_m$') different from the parameters ($p_1$,…, $p_n$, $p_{n+1}$,…, $p_m$). For each parameter, we compute the modules of relative deviations as following:

$$\delta_i = abs((\, p'_i - p_i) / p'_i \,); \;\; i=1,\ldots,m. \tag{2}$$

The absolute deviations ($p'_i$–$p_i$) tend to zero while perturbations tend to zero due to continuous dependence between parameters and perturbations. The same situation is for relative deviations. It given as following:

$$\delta_i \to 0; \; i = 1,\ldots,n; \;\; \text{if} \;\; disturb \to 0. \tag{3}$$

On the contrary, for the unnecessary parameters the values of the relative deviation (2) are close to one due to (1):

$$\delta_i \approx 1; \; i = n+1,\ldots,m; \;\; \text{if} \;\; disturb \neq 0 \tag{4}$$

Criteria (3) and (4) are received for the precision's model. It is possible to extend these criteria to the arbitrary mathematical models.

In the general case, the parameters with low influence on the simulation result have much more relative deviations (2) compared to the required parameters. Consistent elimination of elements of the mathematical model with a high relative deviation leads to improved accuracy and stability of the identification problem. Numerous examples confirm this conclusion [17, 18], including the examples given in this paper.

## 3.2. The reduction algorithm

The following algorithm represents the main stages for the parameters reduction principle. (Algorithm 1).

We propose a mechanical illustration of the reduction principle. The mechanical structure in the form of a bridge truss is in a loaded state. Perturbation of this bridge truss causes fluctuation of

beams. Loaded (necessary) beams will fluctuate with smaller amplitudes than unloaded (unnecessary).

**Algorithm 1.** The parameters reduction

Input data: list of model's parameters, the structure of the model

Output data: reduced list of model's parameters

1. The model identification with parameter $p_i$.
2. Identification of a weakly perturbed model with parameter $p_i$.
3. Calculation of modules of relative deviations of parameters by the formula (2).
4. If there are no relative deviations that are significantly larger than the mean, then the end of the reduction.
5. Element reduction with the biggest δi Go to step 1.

Student's paired t-test for dependent samples is used to check the mean values in the samples. Student's paired t-test is also used to test for a significant difference between the mean values of the accuracy of a subset function in this approach. This is a statistical test that determines the differences between the average values with a certain level of accuracy, assuming that the dependent variable corresponds to the law of normal distribution. It is used in this model to determine the best subset function.

## 4. Results

### 4.1. Lorenz Attractor Test Recovery

The classical equations of the Lorenz attractor (5) are convenient for testing the principle of reduction, since they allow an analytical transformation into an equivalent form (6) convenient for our identification [17].

$$\begin{aligned} &dx_1/dt = 10x_2 - 10x_1; \\ &dx_2/dt = 40x_1 - x_2 - x_1x_3; \\ &dx_3/dt = -8/3 \cdot x_3 + x_1x_2; \\ &x_1(0) = 5;\ \ x_2(0) = 4{,}5;\ \ x_3(0) = 39{,}7. \end{aligned} \tag{5}$$

$$\begin{aligned} &dy_1/dt = y_2; \\ &dy_2/dt = y_3; \\ &dy_3/dt = 1040y_1 - \frac{88}{3}y_2 - \frac{41}{3}y_3 + \\ &+11\frac{y_2^2}{y_1} + \frac{y_2y_3}{y_1} - y_1^2y_2 - 10y_1^3; \\ &y_1(0) = 5;\ \ y_2(0) = -5;\ \ y_3(0) = 20. \end{aligned} \tag{6}$$

So, the problem of reconstruction of the exact model (6) is as follows - having a discrete signal $y_1=x_1$, we must calculate three derivatives of this signal $y_1'=y_2$, $y_2'=y_3$, $y_3'$, and solve the identification problem (7):

$$\min_{\bar{a},\bar{b}} \sum_{m=1}^{M} \left( y'_{3m} - \sum_{i,j,k=0}^{4} a_{ijk} y_{1m}^i y_{2m}^j y_{3m}^k - \sum_{i,j,k=0}^{4} b_{ijk} y_{1m}^i y_{2m}^j y_{3m}^k y_{1m}^{-1} \right)^2 . \tag{7}$$

All polynomial coefficients of the problem (7) are 50. However, for the exact model (6) only 7 coefficients are required. The remaining coefficients are unnecessary.

The discrete signal $y_1=x_1$ is calculated by numerical integration of the equations (6) by the Runge-Kutta method with a step of 0.02 sec. from 0 sec up to 34 sec. On the obtained set of points, an interpolation spline of fifth degree is constructed and its three derivatives are calculated analytically. The numerical values of the spline and its derivatives is the data arrays $y_{1m}$, $y_{2m}$, $y_{3m}$, $y'_{3m}$, ($m$=1,…, 1701) for the identification (7).

Next, the elemental reduction of the arrays of the coefficients $a_{ijk}$, $b_{ijk}$, according to the principle of reduction, was applied. The perturbations were added to values $y'_{3m}$, with a relative value of $10^{-5}$. The relative deviations $\delta_i$ (2) were calculated, and the element with the largest $\delta_i$ was deleted.

The criterion for completing the reduction is the compact set of residual relative deviations. The sign of this is the same number of the relative deviations that are larger and smaller than the average of the remaining area.

In the reduction process, a magnitude of the relative deviations $\max(\delta_i)$-$\min(\delta_i)$, and the middle relative error model coefficients are calculated. After 43 reduction steps, there are 7 coefficients of the exact model (6) with a middle relative error 0.0016. Figure 1 shows a change in the relative error with increasing number of reduction step.

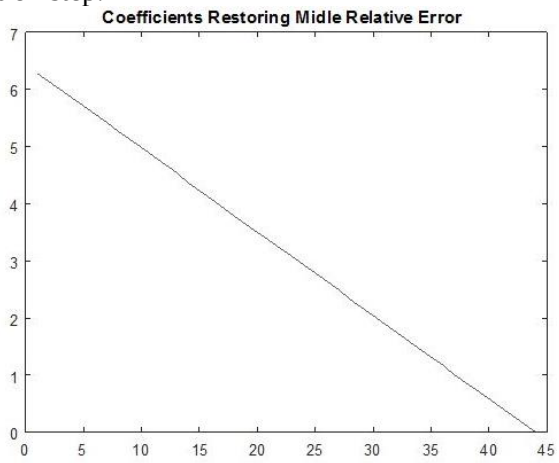


**Figure 1:** Changing the coefficients reproduction relative error of the model (6) in the process of reduction.

The dependence of the area size of relative deviations on the reduction step is shown in Figure 2.

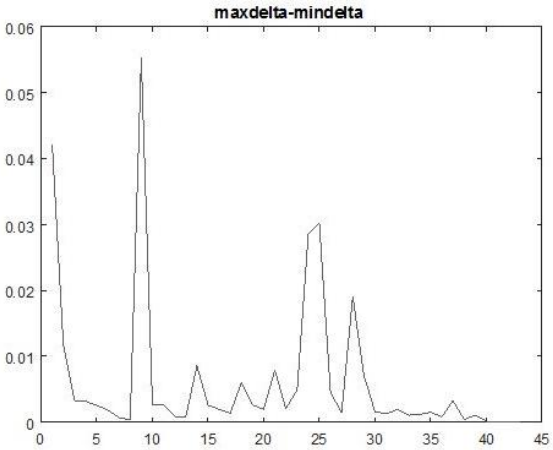


**Figure 2:** Change the size of the area of relative deviations during the reduction.

The same figure on an enlarged scale is shown in Figure 3. The process of forming a compact area is well visible.

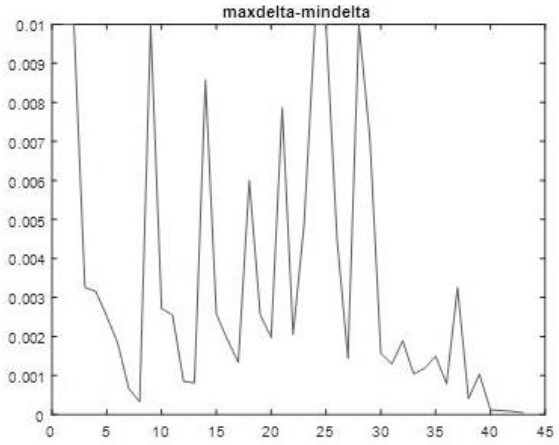


**Figure 3:** The Figure 2 on enlarged scale.

On the Lorentz model, the procedure for increasing (induction) model was tested. Relative deviations (2) for all 50 coefficients were calculated. Next, starting with the three coefficients with the smallest relative deviations, the coefficients with the least relative deviations among the remaining coefficients were successively added to the model. The induction termination criterion is the formation of a compact area of relative deviations in 4 steps.

It is easy to see the advantages of the induction process compared with the reduction. First, the calculation of relative deviations does not need to be repeated at each step. Suffice it to calculate them at the beginning of the induction. Secondly, the number of steps may be smaller.

So, on the test example of the Lorentz attractor reconstruction, the validity of the basic principles of the reduction principle was checked.

## 4.2. Reduction Case for Neural Network

A lot of research [9 – 13] shows the possibility to simplify the structure of neural networks (NN). They are based on the first and second derivatives of the objective function. Opposite to these research, our proposed solution does not depend on the method of network training and type of neural network.

We used the reduction principle for neural network given in [2]. As a result of the reduction of the structure of the neural network model, as a rule, the number of neurons in the input and hidden layers decreases to the optimal number at which the classifying ability of the model will be maximum and at the same time not lower than the initial one (before reduction).

The back propagation is used in the training stage. The network reduction is implemented as a sequential removing of the connections.

The graph of mean square error of approximation of one output variable depending on the number of iteration is shown in Figure 4.

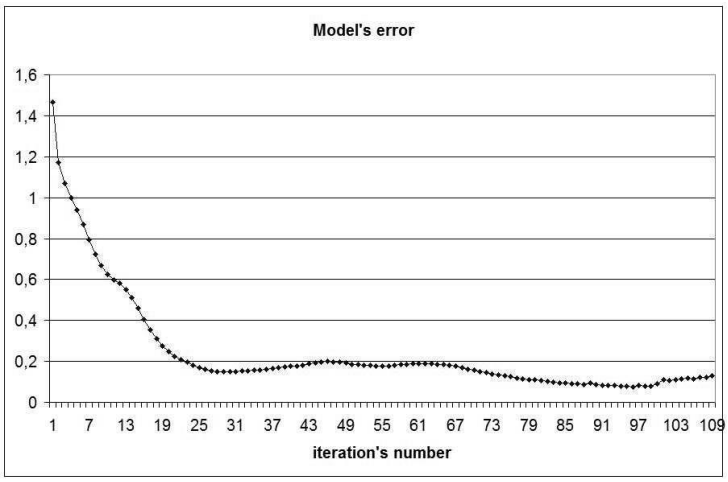


**Figure 4:** The approximation error from the reduction of neural network.

Figure 5 and Figure 6 show the reduction in the range of the relative errors $\delta_i$ values. The relative deviations values in reduced network are assembled in a compact group near zero. This is a sign of the end of the reduction process.

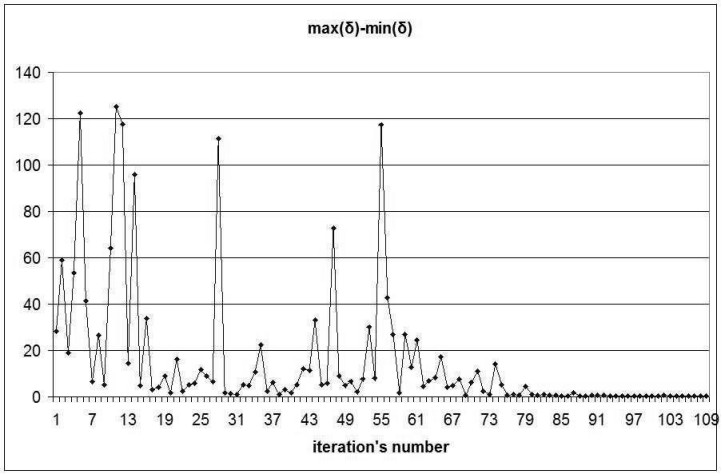


**Figure 5:** The reducing in the range of values of relative errors δi from the reduction of neural network.

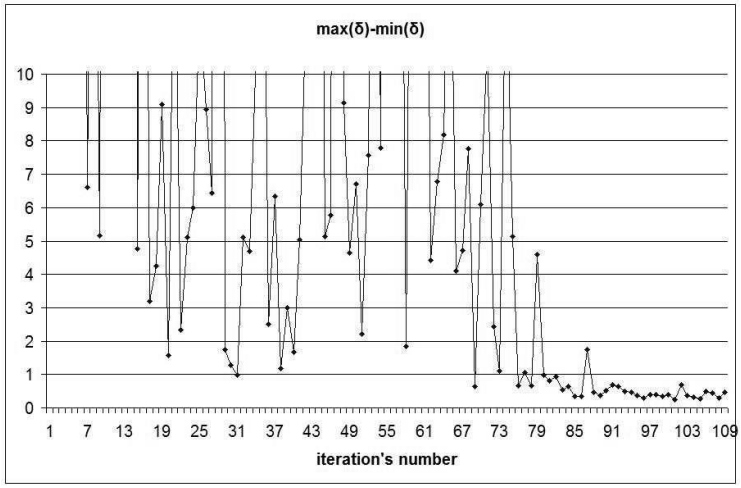


**Figure 6:** The Figure 6 on a larger scale.

We found that the smallest error value is 0.09 on the 99-th iteration, while 35% of connections are removed. The reduction of the network reduced the inaccuracy 16 times.

The next step is estimation of neurons in hidden layers using well-known technics. To determine the number of neurons in the hidden layer of a neural network, it is customary to use a consequence of the Arnold – Kolmogorov – Hecht – Nielsen theorem, according to which the maximum number of neurons in the hidden layer of the perceptron is limited by the right side of expression [13]:

$$Nh < 2\,Nin + 1, \tag{8}$$

where *Nh* is the number of neurons in the hidden layer, and *Nin* is the number of neurons in the input layer.

Thus, this expression determines the upper limit of the number of neurons in the hidden layer of the perceptron neural network model. Consequently, such a number of neurons can lead to redundancy in the structure of the model and, as a consequence, to ineffectiveness of its practical use.

In our case the input architecture of NN satisfies the Arnold – Kolmogorov – Hecht – Nielsen theorem. As results, the standard estimation without error analysis can't optimize the NN.

Let us compare results obtained b Dropout technic. The main idea behind Dropout is instead of training one NN to train an ensemble of several NNs and then average the results. The standard approach implemented in Python 3 shows the average number of disabled neurons is proportional to *np*, where *n* is number of neurons in the initial NN, *p* is Dropout coefficient. (Figure 7).

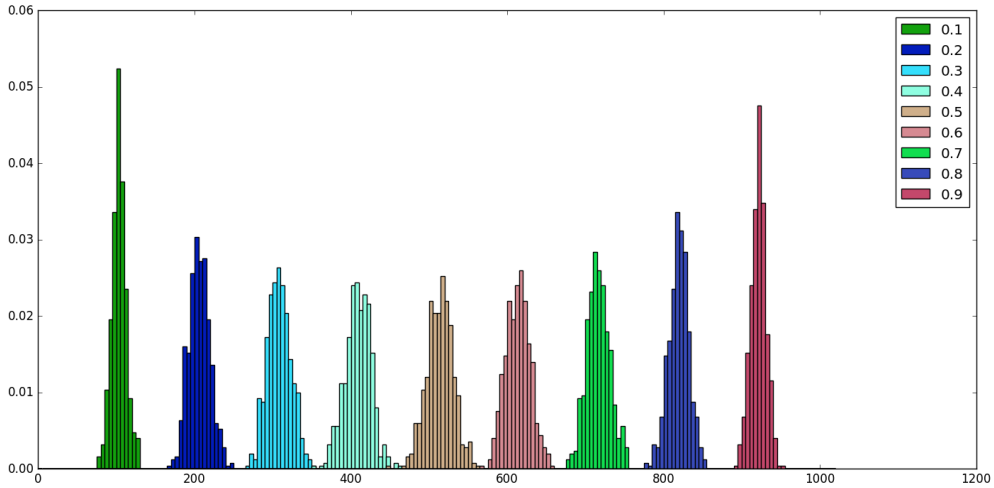


**Figure 7:** The Dropout results for probability *p* from 0 to 1.

The main advantage of the proposed method is ability to evaluate not only the number of removed connections, but also the accuracy of the model. The Dropout technic is based on Rademacher complexity [19]. So, the complexity of the proposed method is much less.

## 4.3. Sun Influence on the Earth's Seismic Activities Model

A crucial task of geo-heliogenic interactions is to model the influence of the Sun on Earth's earthquake and the intensity of near-surface infrasound.

We construct a dynamic model, the variables of which are the intensity of the solar wind *s(k)* during *k*-th day, the average daily earth seismic activity *g(k)*, and the average daily intensity of the near-surface infrasound *z(k)* recorded during *m* days:

$$s(k);\ g(k);\ z(k);\ k=1,\ldots,m. \tag{9}$$

The variable *s(k)* will be the input signal of the model, and the variables *g(k)* and *z(k)* are output signals, and *z(k)* depends on *g(k)*.

The model is chosen as a system of ordinary differential equations:

$$\begin{aligned}
&\dot{s}_i = s_{i+1};\ (i=\overline{0,4});\ \ \dot{s}_5 = P_s(s_0,s_1,s_2,s_3,s_4,s_5);\\
&\dot{g}_i = g_{i+1};\ (i=\overline{0,4});\ \dot{g}_5 = P_g(s_4,s_5,g_0,g_1,g_2,g_3,g_4,g_5);\\
&\dot{z}_i = z_{i+1};\ (i=\overline{0,4});\ \dot{z}_5 = P_z(s_4,s_5,g_4,g_5,z_0,z_1,z_2,z_3,z_4,z_5);
\end{aligned} \tag{10}$$

where the arguments of power polynomials $P_s(.)$, $P_g(.)$ і $P_z(.)$ are selected from a priori considerations.

The general order of the model is 18. Identification of the coefficients of the polynomials is the solution of three problems using the reduction of polynomials

$$\begin{cases} \min\limits_{c_s}\left(\sum\limits_{k=1}^{m}\left[\dot{s}_5(k)-P_s\left(s_0(k),...,s_5(k)\right)\right]^2\right); \\ \min\limits_{c_g}\left(\sum\limits_{k=1}^{m}\left[\dot{g}_5(k)-P_g\left(s_4(k),s_5(k),g_0(k),...,g_5(k)\right)\right]^2\right); \\ \min\limits_{c_z}\left(\sum\limits_{k=1}^{m}\left[\dot{z}_5(k)-P_z\left(s_4(k),s_5(k),g_4(k),g_5(k),z_0(k),...,z_5(k)\right)\right]^2\right), \end{cases} \tag{11}$$

where $c_s$, $c_g$, $c_z$ – coefficients of multidimensional power polynomials $P_s(.)$, $P_g(.)$, $P_z(.)$.

The graphs of the solutions $s_0(t)$, $g_0(t)$, $z_0(t)$ of the model (9), identified by the experimental measurements $s(k)$, $g(k)$, $z(k)$ for 1999, are shown in Figure 8.

The obtained model has high accuracy of reproduction of experimental values only due to reduction. Thus, the relative mean square error of the approximation s(k) model (16), constructed for 119 - 197 days in 1999, is $2.11 \cdot 10^{-4}$. This provides a practical tool for researching and forecasting the activity of simulated geo-heliogenic variables.

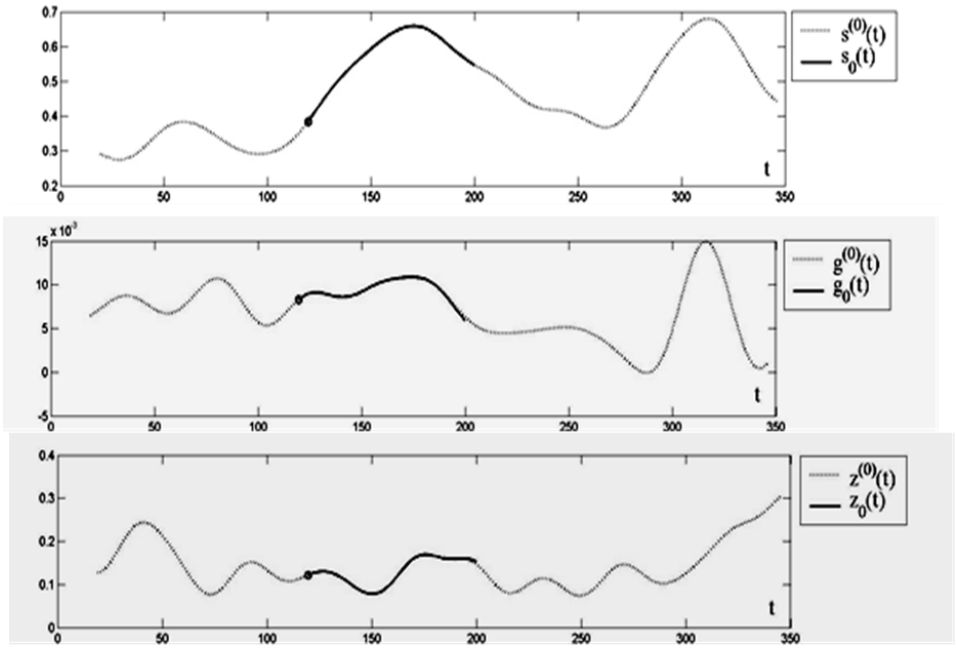


**Figure 8:** Time dependence of solar activity, seismic and infrasound. Experimental dependencies are depicted by lighter lines, and the solutions of the model (11) are darker.

## 5. Conclusions

The novel principle of mathematical model reduction is proposed in the paper. This principle works for mathematical models of any nature, what is explored in this paper on examples.

We analyzed the application for the neural network and the differential equations models. The given examples demonstrate efficient normalizing properties of the reduction principle for the mathematical models in the form of neural networks. The efficiency of the proposed approach comparing to existing methods are verified by different examples explored in the paper. The Dropout technic for neural network reduction is based on Rademacher complexity. So, the complexity of the proposed method is much less.

The results of the developed method can be used in modeling and analysis of the complex systems, particularly for economical modeling [20], energy systems [21].

## 6. References


[1] Melnykova, N., Shakhovska, N., Gregus, M., Melnykov, V., Zakharchuk, M., & Vovk, O. (2020). Data-driven analytics for personalized medical decision making. *Mathematics*, *8*(8), 1211.

[2] Matviychuk, Y., & Matviychuk, O. (2018, April). Reduction principle of the mathematical model and its applications. In *2018 XIV-th International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH)* 174-177.

[3] Becht, Etienne, et al. Dimensionality reduction for visualizing single-cell data using UMAP. Nature biotechnology, 2019, 37.1: 38.

[4] Wang, Rong, et al. Fast and orthogonal locality preserving projections for dimensionality reduction. IEEE Transactions on Image Processing, 2017, 26.10: 5019-5030.

[5] Jamal, Ade, et al. Dimensionality Reduction using PCA and K-Means Clustering for Breast Cancer Prediction. LONTAR KOMPUTER: Jurnal Ilmiah Teknologi Informasi, 2018, 192-201.

[6] Zuo, Z., Wang, C., & Ding, Z. (2017). Robust consensus control of uncertain multi-agent systems with input delay: a model reduction method. International Journal of Robust and Nonlinear Control, 27(11), 1874-1894.

[7] Lee, K., & Carlberg, K. T. (2020). Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. Journal of Computational Physics, 404, 108973.

[8] Swischuk, R., Mainini, L., Peherstorfer, B., & Willcox, K. (2019). Projection-based model reduction: Formulations for physics-based machine learning. Computers & Fluids, 179, 704-717.

[9] He, Y., Zhang, X., & Sun, J. (2017). Channel pruning for accelerating very deep neural networks. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1389-1397).

[10] Louizos, C., Welling, M., & Kingma, D. P. (2017). Learning Sparse Neural Networks through L0 Regularization. arXiv preprint arXiv:1712.01312.

[11] Anderson, D., Ermentrout, B., & Thomas, P. Stochastic representations of ion channel kinetics and exact stochastic simulation of neuronal dynamics. Journal of computational neuroscience, 38(1), 2015, 67-82.

[12] Tkachenko, R., Izonin, I. Model and Principles for the Implementation of Neural-Like Structures based on Geometric Data Transformations. In: Hu, Z.B., Petoukhov, S., (eds) Advances in Computer Science for Engineering and Education. ICCSEEA2018. Advances in Intelligent Systems and Computing. Springer, Cham,754, 2019, 578-587

[13] Izonin I., Tkachenko R., Kryvinska N., Tkachenko P., Greguš ml. M. Multiple Linear Regression Based on Coefficients Identification Using Non-iterative SGTM Neural-like Structure. In: Rojas I., Joya G., Catala A. (eds) Advances in Computational Intelligence. IWANN 2019. Lecture Notes in Computer Science, vol 11506, 2019, 467-479.

[14] Bodyanskiy, Y., Boiko, O., Zaychenko, Y., & Hamidov, G.: Evolving GMDH-neuro-fuzzy system with small number of tuning parameters. In 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD) 2017, 1321-1326

[15] Zhao, C., Ni, B., Zhang, J., Zhao, Q., Zhang, W., & Tian, Q. (2019). Variational convolutional neural network pruning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2780-2789).

[16] Postma, B. N., Poirier-Quinot, D., Meyer, J., & Katz, B. F. (2016). Virtual reality performance auralization in a calibrated model of Notre-Dame Cathedral. Euroregio, 6, 1-10.

[17] Malachivskyy, P. S., Matviychuk, Y. N., Pizyur, Y. V., & Malachivskyi, R. P. (2017). Uniform approximation of functions of two variables. Cybernetics and Systems Analysis, 53(3), 426-431.

[18] Matviychuk, Y., Malachivskyy, P., & Hasko, R. (2012, February). Peculiarities of approximations application to the identification of mathematical macromodels. In Proceedings of International Conference on Modern Problem of Radio Engineering, Telecommunications and Computer Science (pp. 47-48). IEEE

[19] Yin, D., Kannan, R., & Bartlett, P. (2019, May). Rademacher complexity for adversarially robust generalization. In International Conference on Machine Learning (pp. 7085-7094).
[20] Yakovyna, V., Berchenko, N., Kurbanov, K., Virt, I., Kurilo, I., & Nikiforov, Y. (2003). Analysis of the possibility to control complex semiconductors properties by shock wave treatment. *physica status solidi (c)*, (3), 1019-1023.
[21] Kaminskyi, R., Kunanets, N., Rzheuskyi, A., & Khudyi, A. (2018, September). Methods of statistical research for information managers. In *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT)* 2, 127-131.